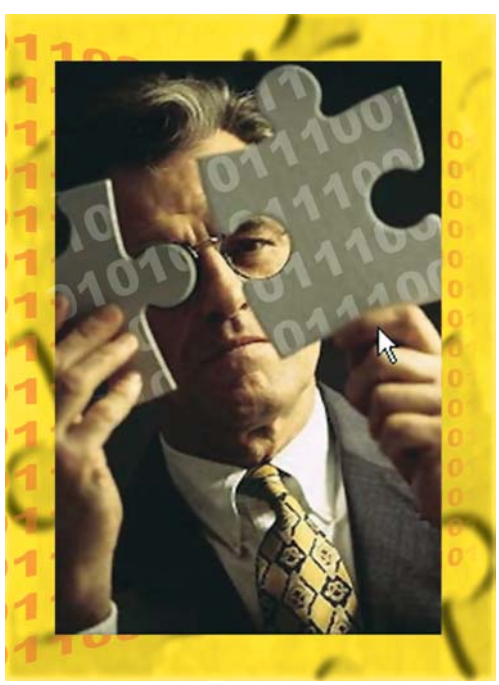
BY: RAJA R.K.

# Managing Log Files in Linux

**Logrotate has been designed to help the system administrator in maintaining files on the production servers to provide high system availability. And it does this pretty well.**

Effective log file management is critical for high system availability. When log files are not properly managed, there is a high possibility of the system running out of disk space and applications crashing often. So it is the responsibility of the system administrator to properly manage all the log files, especially those on the production servers. In Linux, log files can be effectively managed using the freely available Logrotate utility. In this article, you will learn how to manage Linux log files, using the Logrotate utility.



## SETTING UP LOGROTATE

The Logrotate utility will be installed by default when you install any distribution of Linux. To check whether Logrotate has been installed on your system, issue the following command…

```
# rpm -qa | grep -i logrotate
logrotate-3.6.4-1
```

If the utility is not installed on your system, pick either the source or binary version from www.rpmfind.net. You can install the Logrotate RPM package using the following command…

```
# rpm –Uvh <logrotate rpm package name>
```

Table 1 lists various files that are installed as part of the Logrotate RPM package.

***Note:*** You also need to have popt installed in your system to use the Logrotate utility. You can get the popt RPM package from www.rpmfind.net

## LOGROTATE CONFIGURATION

Configuring Logrotate is very simple. You just need to specify the files to be managed in the Logrotate configuration file and configure Logrotate to run as a cron job. Logrotate will manage the files as specified in its configuration file. Any number of configuration files can be passed to the Logrotate utility. The usual practice is to include all the configuration filenames in one configuration file using the *include* directive and passing it to the Logrotate utility. If a directory is specified either in the command line or in the *include* directive, then all the files in that directory will be used as configuration files. You can see a default configuration file logrotate.conf in the /etc directory and a set of default configuration files in the /etc/logrotate.d directory.

A Logrotate configuration file could consist of global and local directives. The global directives apply to all the files specified in the configuration file, and the local directives are file specific. When a directive is specified both in the global and local section, then the latter overrides the former.

Now we will examine the content of the default configuration file /etc/logrotate.conf (installed from logrotate-3.6.4-1 rpm package) in detail. To list the contents of the configuration file, issue the command…

```
# cat /etc/logrotate.conf
weekly
rotate 4
create
include /etc/logrotate.d
/var/log/wtmp {
    monthly
    create 0664 root utmp
    rotate 1
}
```

In the above configuration file, the directives *weekly*, *rotate*, *create* and *include* are global directives, whereas the directives *monthly*, *create* and *rotate* are specific to the /var/log/wtmp file. You can see that the global directives *weekly*, *create* and *rotate* are overridden for the file /var/log/wtmp.

Table 2 describes the various directives that are used in the above default configuration file.

Please see the manual page of Logrotate to know more about the other directives that can be used in the configuration file.

You can check the validity of the Logrotate configuration file as follows…

```
/usr/bin/logrotate -d /etc/
logrotate.conf
```

The above command will run the Logrotate utility in dry run mode.

Once you are done with the configuration file, the next step is to configure Logrotate to run as cron job. All that you need to do is…

- Create a crontab file name /etc/logrotate.crontab and add the following line…

```
0 22 * * * /usr/sbin/logrotate /etc/
logrotate.conf
```

The above crontab file will instruct the cron daemon to run Logrotate daily at 10 P.M.

- Install the crontab file…

```
crontab /etc/logrotate.crontab
```

## MORE CONFIGURATION EXAMPLES

The below given configuration file instructs Logrotate to use /bin/gzip to compress the /var/log/test.log file. The *copy* directive instructs the Logrotate to make a copy of /var/log/test.log without modifying it…

```
compresscmd /bin/gzip
compress
rotate 3

/var/log/test.log {
# Take a copy of the original file
without modifying it
copy
}
```

The following commands show Logrotate in action…

```
# ls -ltri /var/log/test*
32276 -rw-r--r--    1 root     root
4764 Jan  7 09:49 /var/log/test.log

# /usr/sbin/logrotate -f /etc/
logrotate.conf

# ls -ltri /var/log/test*
32276 -rw-r--r--    1 root     root
4764 Jan  7 09:49 /var/log/test.log
32282 -rw-r--r--    1 root     root
1086 Jan  7 09:49 /var/log/test.log.1.gz
```

In the above example, 32276 and 32282 denote the inode number of /var/log/test.log and /var/log/test.log.1.gz respectively.

In the configuration file given below, the *create* directive instructs the Logrotate to first back up (move) the original file and then create a new one with the same name. When used, it is the responsibility of the user to inform the programs that are writing to /var/log/test.log about the rotation so that they can reopen the file, else they might end up writing to the backed up (old) file.

For programs that cannot be informed about the rotation, use *copytruncate* instead of *create*.

```
compresscmd /bin/gzip
compress
rotate 3
/var/log/test.log {
# Backup the original file and then
create a new one with same name
create
}
```

The following commands show Logrotate in action…

```
# ls -ltri /var/log/test*
32276 -rw-r--r--    1 root     root
0 Jan  7 09:51 /var/log/test.log

# logrotate -f /etc/logrotate.conf

#ls -ltri /var/log/test*
32282 -rw-r--r--    1 root     root
31 Jan  7 09:51 /var/log/test.log.1.gz
```

**Table 1**

| File | Description |
|---|---|
| /etc/cron.daily/logrotate | Default cron job file |
| /etc/logrotate.conf | Default Logrotate configuration file |
| /etc/logrotate.d | Directory containing default Logrotate configuration files (see include directive) |
| /usr/sbin/logrotate | Logrotate executable |
| /usr/share/doc/logrotate-3.6.4 /usr/share/doc/logrotate-3.6.4/CHANGES/ usr/share/man/man8/logrotate.8.gz | Documentation |
| /var/lib/logrotate.status | Status file (will hold the logrotation status) |

**Table 2**

| Directive | Description |
|---|---|
| Weekly | Instructs the Logrotate to manage the files once in a week |
| rotate 4 | Instructs the Logrotate to keep up to four versions of the old file |
| Create | Instructs the Logrotate to first move the old logfile and then create a new one with the same name as the original log file. *Example:* If the filename is wtmp, then the logrotate will first move wtmp to wtmp.1 and create a new file wtmp |
| Include | Instructs the logrotate to treat all the files under /etc/logrotate.d as configuration files |

```
32277 -rw-r—r—    1 root     root
0 Jan  7 10:01 /var/log/test.log
```

Note the inode number of /var/log/test.log before and after the rotation.

In the following configuration file the *copytruncate* directive can be used when the program that is writing to /var/log/test.log cannot be informed about the rotation (change in the file) and so might continue writing to the old log file…

```
compresscmd /bin/gzip
compress
rotate 3
/var/log/test.log {
# Take a copy of the original file and
then truncate the original file
copytruncate
}
```

The following commands show the Logrotate in action…

```
# ls -ltri /var/log/test*
32276 -rw-r—r—    1 root     root
4764 Jan  7 09:49 /var/log/test.log

# logrotate -f /etc/logrotate.conf

# ls -ltri /var/log/test*
32276 -rw-r—r—    1 root     root
4764 Jan  7 09:49 /var/log/test.log
32282 -rw-r—r—    1 root     root
1086 Jan  7 09:51 /var/log/test.log.1.gz
```

Note the inode number of the /var/log/test.log before and after rotation.

In the configuration file given below, the *missingok* directive instructs the Logrotate to move on to the next file in the configuration if either the /var/log/test.log or /var/log/another_file.log are missing without issuing an error. The *notifempty* directive instructs Logrotate to not to rotate the file if it is empty. The *size* directive instructs the Logrotate to rotate the files if their size exceeds 100 MB. The *sharedscripts* directive instructs the Logrotate to execute the prerotate and postrotate scripts only once. The sections between *prerotate/endscript* and *postrotate/endscript* are executed before and after rotation…

```
compresscmd /bin/gzip
compress
rotate 3
/var/log/test.log /var/log/
another_file.log {
        missingok
        notifempty
                size=100M
        sharedscripts
        prerotate
                        echo
“Before rotation” | mail -s
“logrotation” root
        endscript
        postrotate
                echo “After rotation”
| mail -s “logrotation” root
kill -SIGHUP <pid of the program that is
writing to /var/log/test.log and /var/
log/another_file.log>
        endscript
create 0775 root root
}
```

The following commands show Logrotate in action…

```
# ls -ltri /var/log/test* /var/log/
another*
32277 -rwxrwxr-x    1 root     root
4852 Jan  7 10:47 /var/log/
another_file.log
32282 -rwxrwxr-x    1 root     root
4852 Jan  7 10:47 /var/log/test.log

# logrotate -f /etc/logrotate.conf

# ls -ltri /var/log/test* /var/log/
another*
```

```
32284 -rwxrwxr-x    1 root     root
1110 Jan  7 10:47 /var/log/
another_file.log.1.gz
32283 -rwxrwxr-x    1 root     root
1102 Jan  7 10:47 /var/log/test.log.1.gz
32276 -rwxrwxr-x    1 root     root
0 Jan  7 10:47 /var/log/test.log
32282 -rwxrwxr-x    1 root     root
0 Jan  7 10:47 /var/log/another_file.log

# mail
Mail version 8.1 6/6/93.  Type ? for
help.
“/var/spool/mail/root”: 177 messages 2
new 177 unread
>N176 root@localhost.local  Wed Jan  7
10:47  13/418   “logrotation”
 N177 root@localhost.local  Wed Jan  7
10:47  13/419   “logrotation”
& n
Message 176:
From root  Wed Jan  7 10:47:21 2004
Date: Wed, 7 Jan 2004 10:47:21 +0530
From: root <root@localhost.localdomain>
To: root@localhost.localdomain
Subject: logrotation

After rotation

& n
Message 177:
From root  Wed Jan  7 10:47:21 2004
Date: Wed, 7 Jan 2004 10:47:21 +0530
From: root <root@localhost.localdomain>
To: root@localhost.localdomain
Subject: logrotation

Before rotation

& n
At EOF
```

LFY

*The author is a lead engineer working in HCL Technologies (Cisco Systems Offshore Development Centre) in Chennai. He has more than five years of experience in software development including two years in Linux. He holds a B.E. degree in computer science & engineering from S.R.M Engineering College. He can be contacted at rajark_hcl@yahoo.co.in*